Presentation

Agenda

HPCS HPCchallenge Benchmark Suite

David Koester
The MITRE Corporation
Email Address: dkoester@mitre.org

Jack Dongarra and ***Piotr Luszczek***
ICL/UT
Email Addresses: {dongarra, luszczek}@cs.utk.edu

Abstract

The Defense Advanced Research Projects Agency (DARPA) High Productivity Computing Systems (HPCS) HPCchallenge Benchmarks examine the performance of High Performance Computing (HPC) architectures using kernels with more challenging memory access patterns than just the High Performance LINPACK (HPL) benchmark used in the Top500 list. The HPCchallenge Benchmarks build on the HPL framework and augment the Top500 list by providing benchmarks that bound the performance of many real applications as a function of memory access locality characteristics. The real utility of the HPCchallenge benchmarks are that architectures can be described with a wider range of metrics than just Flop/s from HPL. Even a small percentage of random memory accesses in real applications can significantly affect the overall performance of that application on architectures not designed to minimize or hide memory latency. The HPCchallenge Benchmarks includes a new metric — Giga UPdates per Second — and a new benchmark — RandomAccess — to measure the ability of an architecture to access memory randomly, i.e., with no locality. When looking only at HPL performance and the Top500 List, inexpensive build-your-own clusters appear to be much more cost effective than more sophisticated HPC architectures. HPCchallenge Benchmarks provide users with additional information to justify policy and purchasing decisions. We will compare the measured HPCchallenge Benchmark performance on various HPC architectures — from Cray X1s to Beowulf clusters — in the presentation and paper. Additional information on the HPCchallenge Benchmarks can be found at http://icl.cs.utk.edu/hpcc/

Introduction

At SC2003 in Phoenix (15-21 November 2003), Jack Dongarra (ICL/UT) announced the release of a new benchmark suite — the HPCchallenge Benchmarks — that examine the performance of HPC architectures using kernels with more *challenging* memory access patterns than High Performance Linpack (HPL) used in the Top500 list. The HPCchallenge Benchmarks are being designed to *complement* the Top500 list and provide benchmarks that *bound* the performance of many real applications as a function of memory access characteristics — e.g., spatial and temporal locality. Development of the HPCchallenge Benchmarks is being funded by the Defense Advanced Research Projects Agency (DARPA) High Productivity Computing Systems (HPCS) Program.

The HPCchallenge Benchmark Kernels

Local	**Global**
DGEMM (matrix x matrix multiply)	High Performance LINPACK (HPL)
STREAM • COPY • SCALE • ADD • TRIADD	PTRANS — parallel matrix transpose
RandomAccess	(MPI)RandomAccess
1D FFT	1D FFT
I/O b_eff — effective bandwidth benchmark	

Report Documentation Page	*Form Approved* *OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **01 FEB 2005**	2. REPORT TYPE **N/A**	3. DATES COVERED **-**

4. TITLE AND SUBTITLE **HPCS HPCchallenge Benchmark Suite**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **The MITRE Corporation**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S)
	11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES

See also ADM00001742, HPEC-7 Volume 1, Proceedings of the Eighth Annual High Performance Embedded Computing (HPEC) Workshops, 28-30 September 2004 Volume 1., The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**UU**	**28**	

Additional information on the HPCchallenge Benchmarks can be found at http://icl.cs.utk.edu/hpcc/.

Flop/s

The Flop/s metric from HPL has been the de facto standard for comparing High Performance Computers for many years. HPL works well on all architectures — even cache-based, distributed memory multiprocessors — and the measured performance may not be representative of a wide range of real user applications like adaptive multi-physics simulations used in weapons and vehicle design and weather, climate models, and defense applications. HPL is more compute friendly than these applications because it has more extensive memory reuse in the Level 3 BLAS-based calculations. .

Memory Performance

There is a need for benchmarks that test memory performance. When looking only at HPL performance and the Top500 List, inexpensive build-your-own clusters appear to be much more cost effective than more sophisticated HPC architectures. HPL has high spatial and temporal locality — characteristics shared by few real user applications. HPCchallenge benchmarks provide users with additional information to justify policy and purchasing decisions

Not only does the Japanese Earth Simulator outperform the top American systems on the HPL benchmark (Tflop/s), the differences in bandwidth performance on John McCalpin's STREAM TRIAD benchmark (Level 1 BLAS) shows even greater performance disparity. The Earth Simulator outperforms the ASCI Q by a factor of 4.64 on HPL. Meanwhile, the higher bandwidth memory and interconnect systems of the Earth Simulator are clearly evident as it outperforms ASCI Q by a factor of 36.25 on STREAM TRIAD. In the presentation and paper, we will compare the measured HPCchallenge Benchmark performance on various HPC architectures — from Cray X1s to Beowulf clusters — using the updated results at http://icl.cs.utk.edu/hpcc/hpcc_results.cgi

Even a small percentage of random memory accesses in real applications can significantly affect the overall performance of that application on architectures not designed to minimize or hide memory latency. Memory latency has not kept up with Moore's Law. Moore's Law hypothesizes a 60% compound growth rate per year for microprocessor "performance", while memory latency has been improving at a compound rate of only 7% per year. The memory-processor performance gap has been growing at a rate of over 50% per year since 1980. The HPCchallenge Benchmarks includes a new metric — Giga UPdates per Second — and a new benchmark — RandomAccess — to measure the ability of an architecture to access memory randomly, i.e., with no locality.

GUPS is calculated by identifying the number of memory locations that can be randomly updated in one second, divided by 1 billion (1e9). The term "randomly" means that there is little relationship between one address to be updated and the next, except that they occur in the space of ½ the total system memory. An update is a read-modify-write operation on a table of 64-bit words. An address is generated, the value at that address read from memory, modified by an integer operation (add, and, or, xor) with a literal value, and that new value is written back to memory

Abstract Agenda

HPCS HPCchallenge Benchmark Suite

David Koester, Ph.D. (MITRE)
Jack Dongarra (UTK)
Piotr Luszczek (ICL/UTK)

28 September 2004

Outline

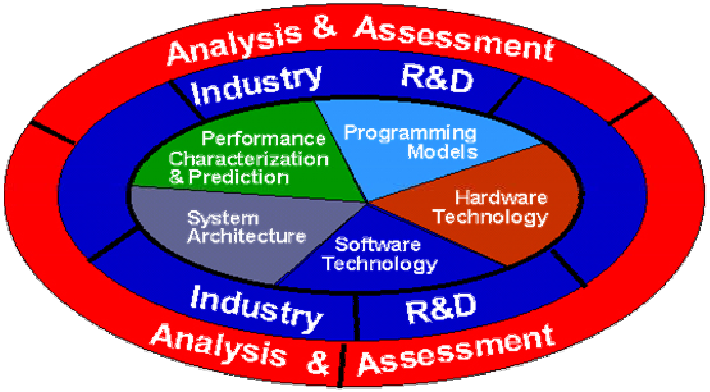
- Brief DARPA HPCS Overview
- Architecture/Application Characterization
- HPCchallenge Benchmarks
- Preliminary Results
- Summary

High Productivity Computing Systems

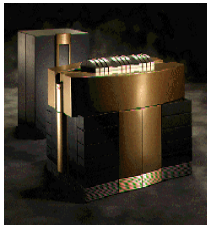
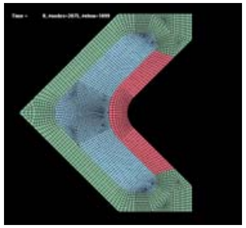
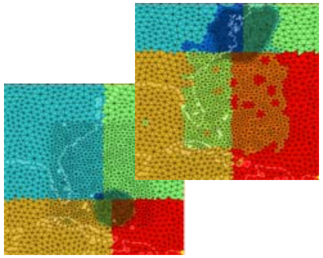
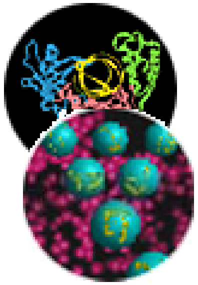
- **Create a new generation of economically viable computing systems and a procurement methodology for the security/industrial community (2007 – 2010)**

Impact:

- **Performance** (time-to-solution): speedup critical national security applications by a factor of 10X to 40X
- **Programmability** (idea-to-first-solution): reduce cost and time of developing application solutions
- **Portability** (transparency): insulate research and operational application software from system
- **Robustness** (reliability): apply all known techniques to **protect against outside attacks**, hardware faults, & programming errors



HPCS Program Focus Areas

Applications:

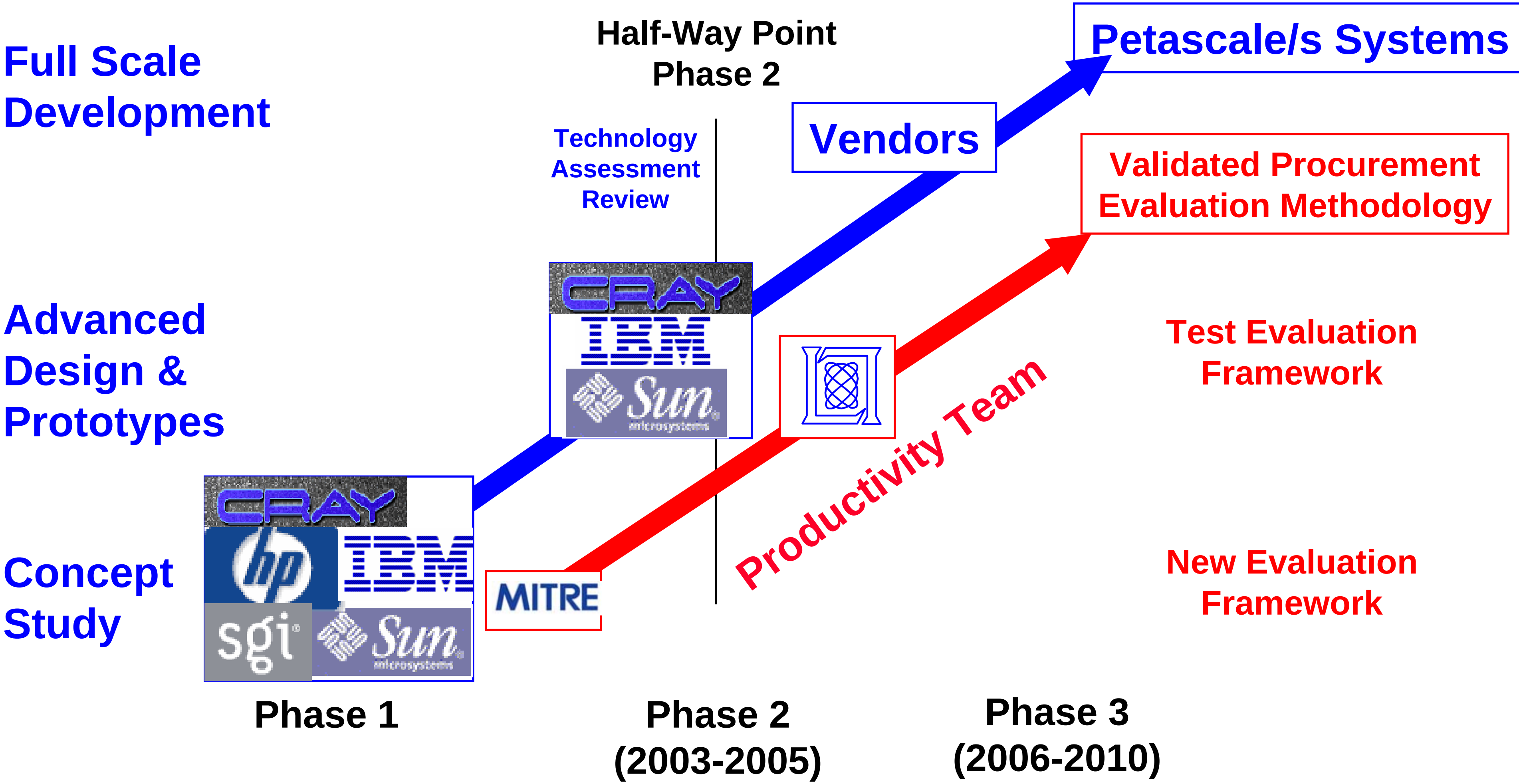
- Intelligence/surveillance, reconnaissance, cryptanalysis, weapons analysis, airborne contaminant modeling and biotechnology

Fill the Critical Technology and Capability Gap

Today (late 80's HPC technology)…..*to*…..Future (Quantum/Bio Computing)

High Productivity Computing Systems
-Program Overview-

- Create a new generation of economically viable computing systems and a procurement methodology for the security/industrial community (2007 – 2010)



Full Scale Development

Advanced Design & Prototypes

Concept Study

Half-Way Point Phase 2

Technology Assessment Review

Petascale/s Systems

Vendors

Validated Procurement Evaluation Methodology

Test Evaluation Framework

New Evaluation Framework

Productivity Team

Phase 1

Phase 2 (2003-2005)

Phase 3 (2006-2010)

HPCS Program Goals[‡]

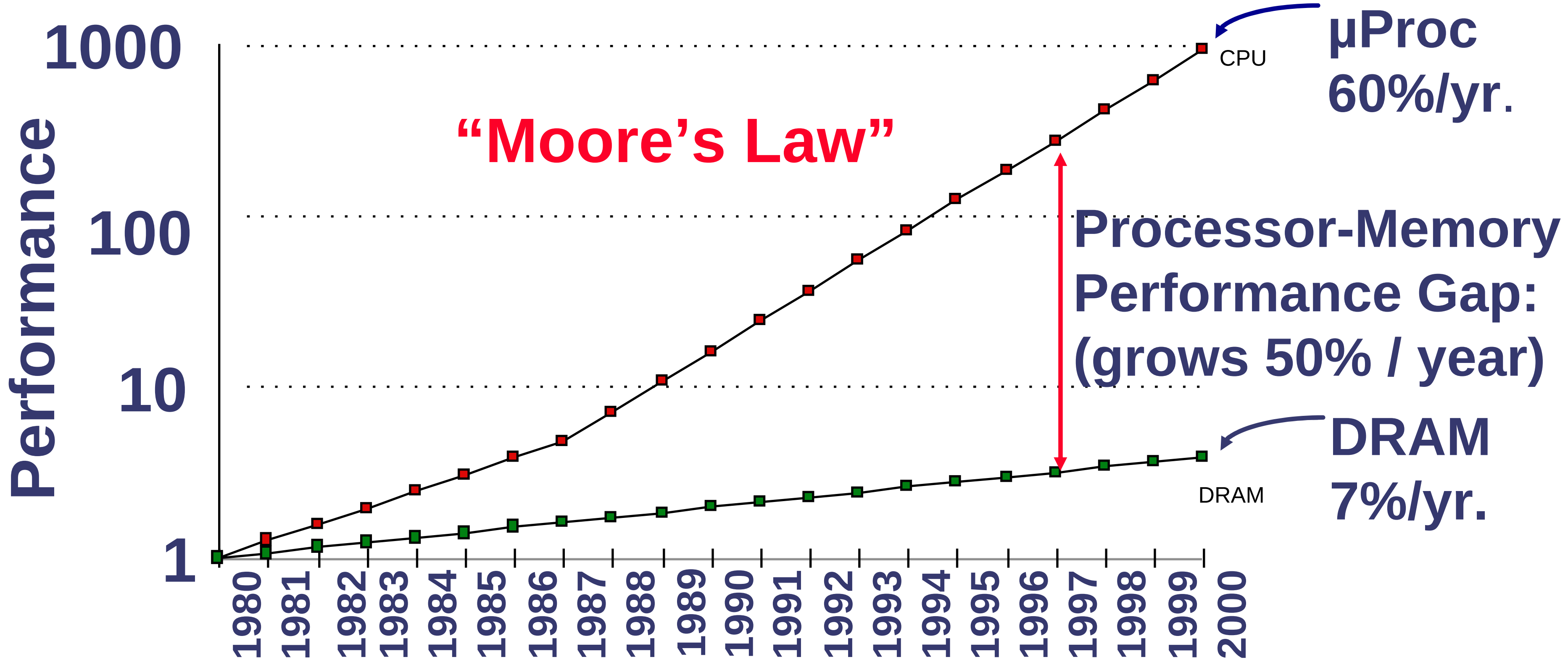
- **HPCS overall productivity goals:**
  - Execution (sustained performance)
    - 1 Petaflop/sec (scalable to greater than 4 Petaflop/sec)
    - Reference: Production workflow
  - Development
    - 10X over today's systems
    - Reference: Lone researcher and Enterprise workflows
- **Productivity Framework**
  - Base lined for today's systems
  - Successfully used to evaluate the vendors emerging productivity techniques
  - Provide a solid reference for evaluation of vendor's proposed Phase III designs.
- **Subsystem Performance Indicators**
  1) 2+ PF/s LINPACK
  2) 6.5 PB/sec data STREAM bandwidth
  3) 3.2 PB/sec bisection bandwidth
  4) 64,000 GUPS

[‡] Bob Graybill (DARPA/IPTO)
(Emphasis added)

Outline

- Brief DARPA HPCS Overview
- **Architecture/Application Characterization**
- **HPCchallenge Benchmarks**
- **Preliminary Results**
- **Summary**

Processor-Memory Performance Gap



- •Alpha 21264 full cache miss / instructions executed: 180 ns/1.7 ns =108 clks x 4 or 432 instructions
- • Caches in Pentium Pro: 64% area, 88% transistors

*Taken from Patterson-Keeton Talk to SigMod

Processing vs. Memory Access

- Doesn’t cache solve this problem?
  - It depends. With small amounts of contiguous data, usually. With large amounts of non-contiguous data, usually not
  - In most computers the programmer has no control over cache
  - Often “a few” Bytes/FLOP is considered OK

- However, consider operations on the transpose of a matrix (e.g., for adjunct problems)
  - Xa= b $\qquad$ $X^Ta = b$
  - If X is big enough, 100% cache misses are guaranteed, and we need at least 8 Bytes/FLOP (assuming a and b can be held in cache)

- Latency and limited bandwidth of processor-memory and node-node communications are major limiters of performance for scientific computation

Processing vs. Memory Access
High Performance LINPACK

Consider another benchmark: Linpack

$A x = b$

Solve this linear equation for the vector x, where A is a known matrix, and b is a known vector. Linpack uses the BLAS routines, which divide A into blocks.

On the average Linpack requires 1 memory reference for every 2 FLOPs, or 4Bytes/Flop.

Many of these can be cache references

Processing vs. Memory Access STREAM TRIAD

Consider the simple benchmark: STREAM TRIAD

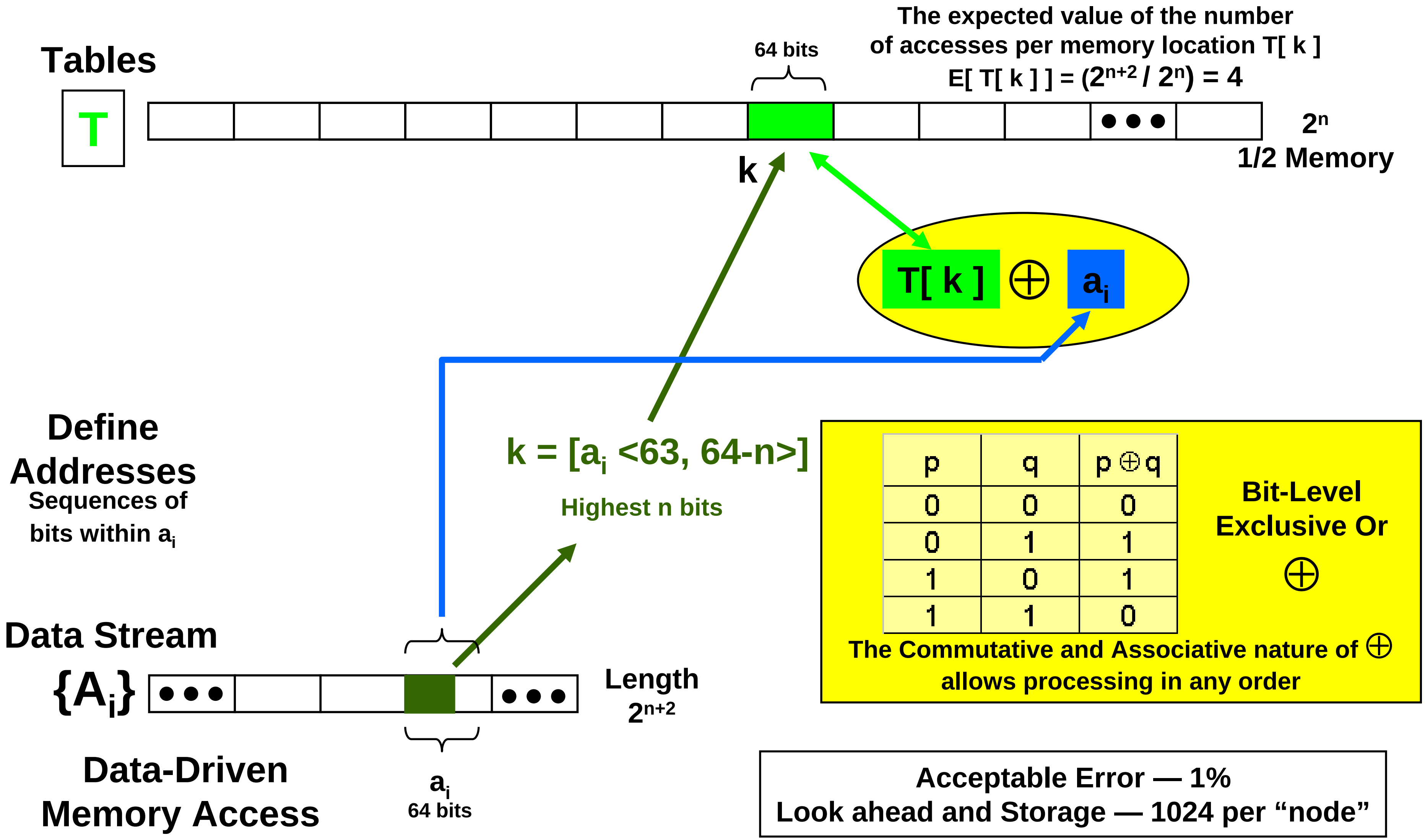
```
a(i) = b(i) + q * c(i)
```

`a(i)`, `b(i)`, and `c(i)` are vectors; q is a scalar
Vector length is chosen to be much longer than cache size

Each execution includes
2 memory loads + 1 memory store
2 FLOPs
12 Bytes/FLOP (assuming 32 bit precision)

No computer has enough memory bandwidth to reference
12 Bytes for each FLOP!

Processing vs. Memory Access RandomAccess



Tables

T

64 bits

The expected value of the number of accesses per memory location T[k]

E[T[k]] = $(2^{n+2} / 2^n) = 4$

● ● ●

2^n

1/2 Memory

k

T[k] ⊕ a_i

Define Addresses

Sequences of bits within a_i

k = [a_i <63, 64-n>]

Highest n bits

p	q	p ⊕ q
0	0	0
0	1	1
1	0	1
1	1	0

Bit-Level Exclusive Or

⊕

The Commutative and Associative nature of ⊕ allows processing in any order

Data Stream

$\{A_i\}$

● ● ● ● ● ●

Length 2^{n+2}

Data-Driven Memory Access

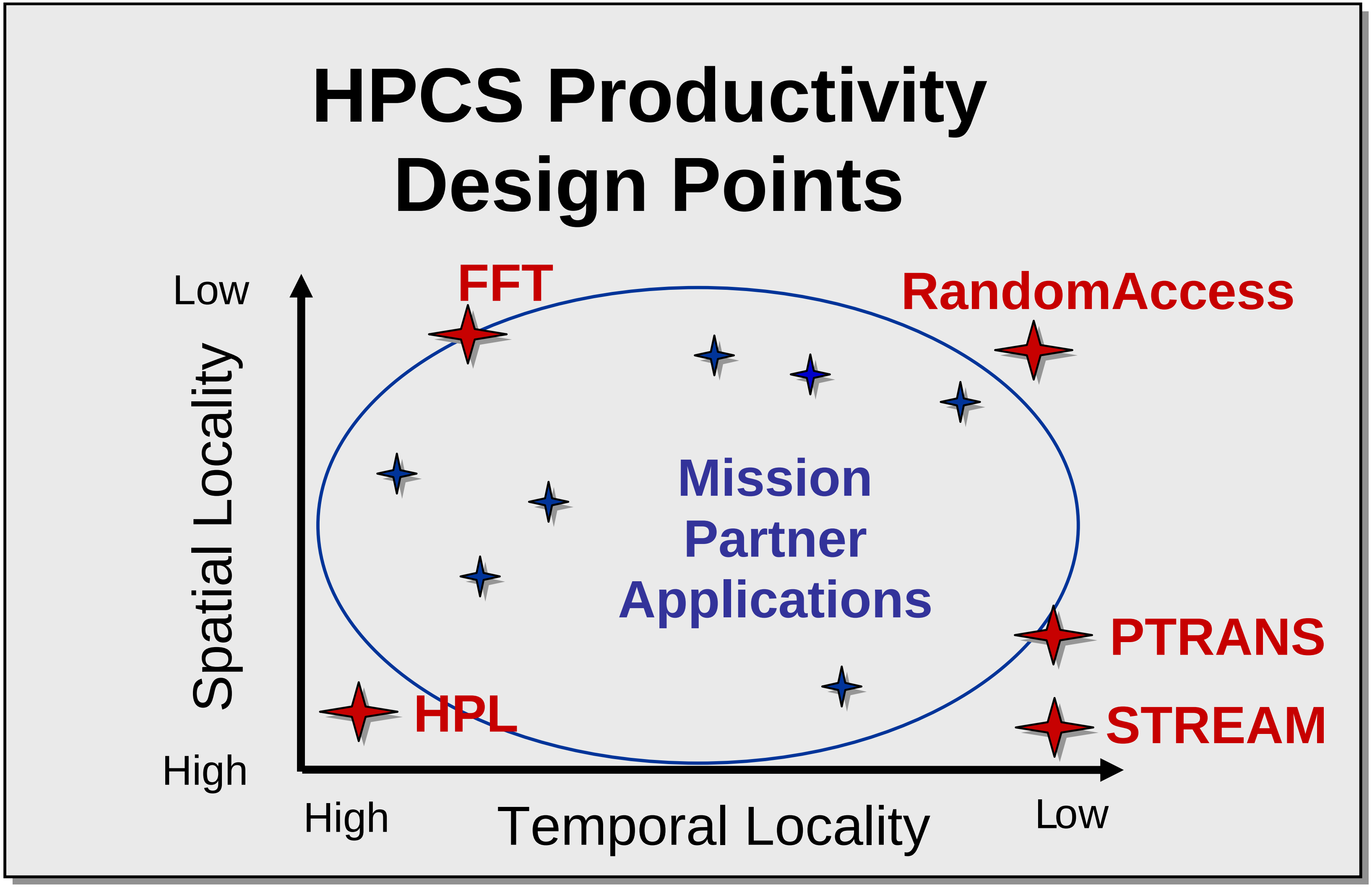
a_i

64 bits

Acceptable Error — 1%

Look ahead and Storage — 1024 per “node”

Bounding Mission Partner Applications



Outline

- Brief DARPA HPCS Overview
- Architecture/Application Characterization
- HPCchallenge Benchmarks
- Preliminary Results
- Summary

HPCS HPCchallenge Benchmarks

- **HPCSchallenge Benchmarks**
  - Being developed by Jack Dongarra (ICL/UT)
  - Funded by the DARPA High Productivity Computing Systems (HPCS) program (Bob Graybill (DARPA/IPTO))

To examine the performance of High Performance Computer (HPC) architectures using kernels with more *challenging* memory access patterns than High Performance Linpack (HPL)

HPCchallenge Goals

- To examine the performance of HPC architectures using kernels with more *challenging* memory access patterns than HPL
  - HPL works well on all architectures — even cache-based, distributed memory multiprocessors due to
    1. Extensive memory reuse
    2. Scalable with respect to the amount of computation
    3. Scalable with respect to the communication volume
    4. Extensive optimization of the software
- To *complement* the Top500 list
- To provide benchmarks that *bound* the performance of many real applications as a function of memory access characteristics — e.g., spatial and temporal locality

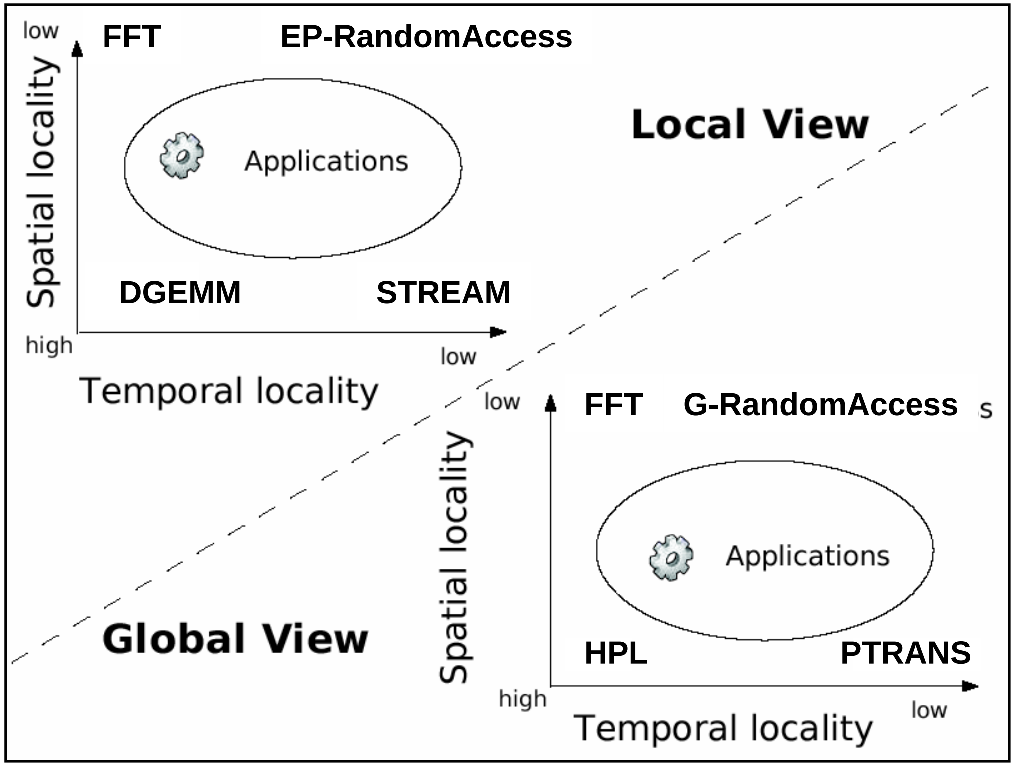
HPCchallenge Benchmarks

Local

- DGEMM (matrix x matrix multiply)
- STREAM
  - COPY
  - SCALE
  - ADD
  - TRIADD
- EP-RandomAccess
- 1D FFT

Global

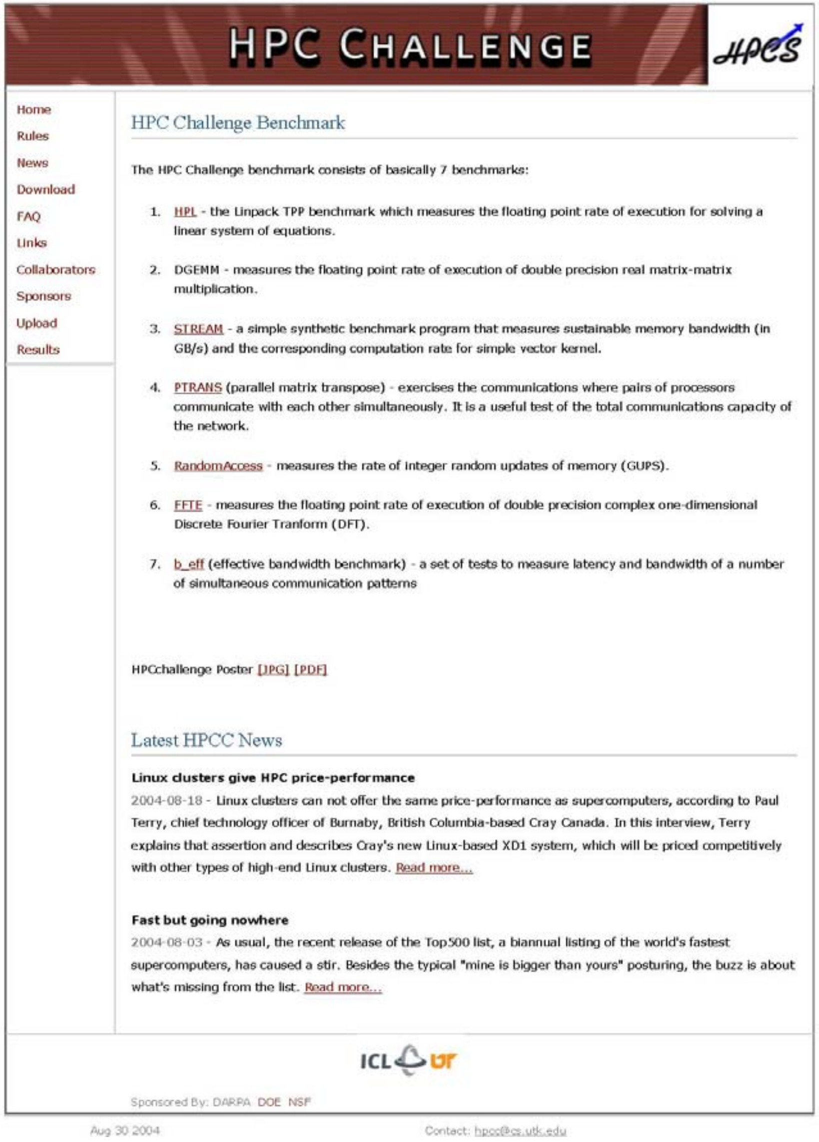
- High Performance LINPACK (HPL)
- PTRANS — parallel matrix transpose
- G-RandomAccess
- 1D FFT
- b_eff — interprocessor bandwidth and latency



- HPCchallenge pushes spatial and temporal boundaries; sets performance bounds
- Available for download http://icl.cs.utk.edu/hpcc/

Web Site
http://icl.cs.utk.edu/hpcc/

- Home
- Rules
- News
- Download
- FAQ
- Links
- Collaborators
- Sponsors
- Upload
- Results



HPC Challenge Benchmark

The HPC Challenge benchmark consists of basically 7 benchmarks:

1. HPL - the Linpack TPP benchmark which measures the floating point rate of execution for solving a linear system of equations.
2. DGEMM - measures the floating point rate of execution of double precision real matrix-matrix multiplication.
3. STREAM - a simple synthetic benchmark program that measures sustainable memory bandwidth (in GB/s) and the corresponding computation rate for simple vector kernel.
4. PTRANS (parallel matrix transpose) - exercises the communications where pairs of processors communicate with each other simultaneously. It is a useful test of the total communications capacity of the network.
5. RandomAccess - measures the rate of integer random updates of memory (GUPS).
6. FFTE - measures the floating point rate of execution of double precision complex one-dimensional Discrete Fourier Tranform (DFT).
7. b_eff (effective bandwidth benchmark) - a set of tests to measure latency and bandwidth of a number of simultaneous communication patterns

HPCchallenge Poster [JPG] [PDF]

Latest HPCC News

Linux clusters give HPC price-performance

2004-08-18 - Linux clusters can not offer the same price-performance as supercomputers, according to Paul Terry, chief technology officer of Burnaby, British Columbia-based Cray Canada. In this interview, Terry explains that assertion and describes Cray's new Linux-based XD1 system, which will be priced competitively with other types of high-end Linux clusters. Read more...

Fast but going nowhere

2004-08-03 - As usual, the recent release of the Top500 list, a biannual listing of the world's fastest supercomputers, has caused a stir. Besides the typical "mine is bigger than yours" posturing, the buzz is about what's missing from the list. Read more...

Sponsored By: DARPA DOE NSF

Aug 30 2004

Contact: hpcc@cs.utk.edu

Outline

- Brief DARPA HPCS Overview
- Architecture/Application Characterization
- HPCchallenge Benchmarks
- Preliminary Results
- Summary

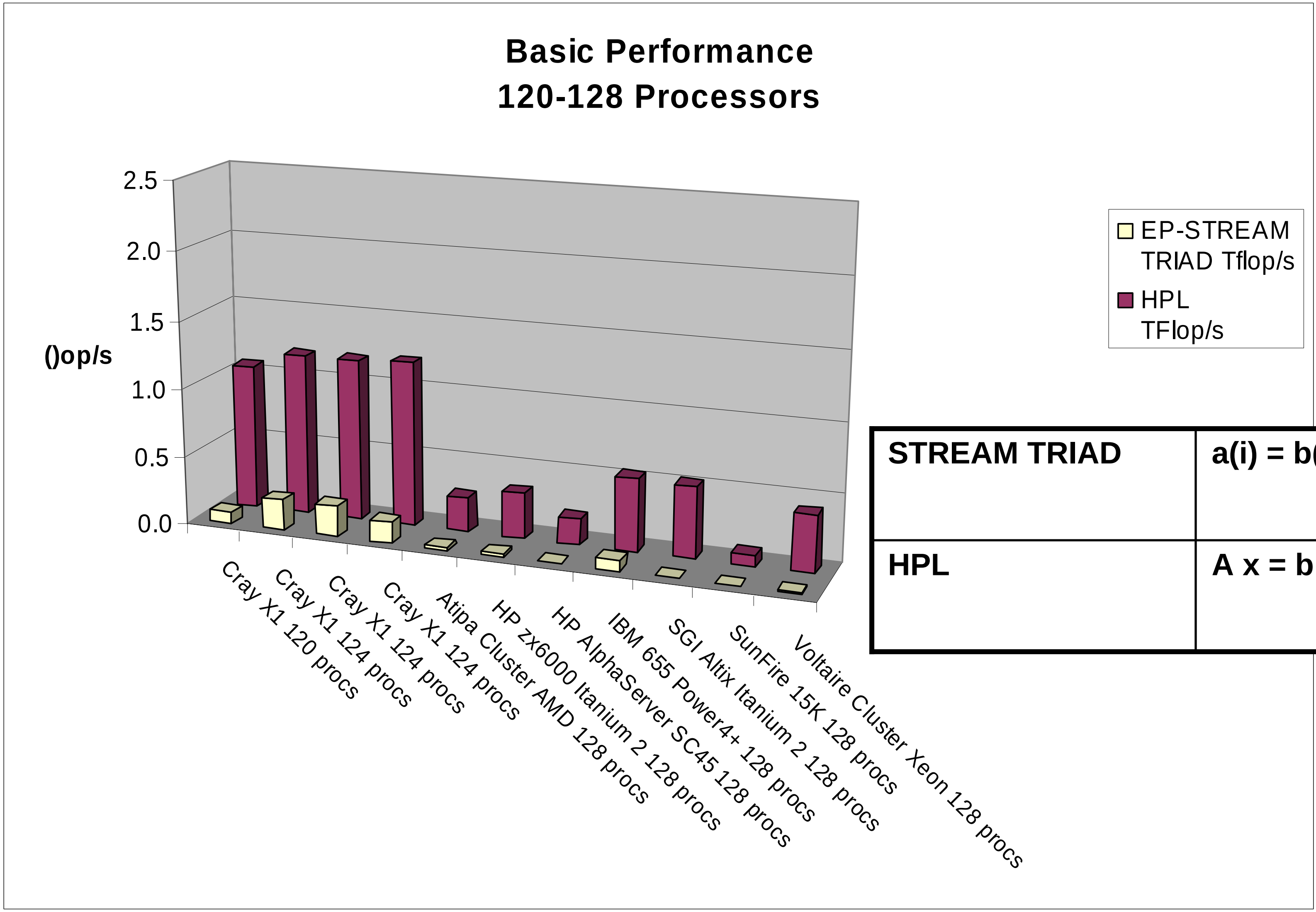
Preliminary Results Machine List (1 of 2)

Affiliation	Manufacturer	System	ProcessorType	Procs
U Tenn	Atipa Cluster AMD 128 procs	Conquest cluster	AMD Opteron	128
AHPCRC	Cray X1 124 procs	X1	Cray X1 MSP	124
AHPCRC	Cray X1 124 procs	X1	Cray X1 MSP	124
AHPCRC	Cray X1 124 procs	X1	Cray X1 MSP	124
ERDC	Cray X1 60 procs	X1	Cray X1 MSP	60
ERDC	Cray X1 60 procs	X1	Cray X1 MSP	60
ORNL	Cray X1 252 procs	X1	Cray X1 MSP	252
ORNL	Cray X1 252 procs	X1	Cray X1 MSP	252
AHPCRC	Cray X1 120 procs	X1	Cray X1 MSP	120
ORNL	Cray X1 64 procs	X1	Cray X1 MSP	64
AHPCRC	Cray T3E 1024 procs	T3E	Alpha 21164	1024
ORNL	HP zx6000 Itanium 2 128 procs	Integrity zx6000	Intel Itanium 2	128
PSC	HP AlphaServer SC45 128 procs	AlphaServer SC45	Alpha 21264B	128
ERDC	HP AlphaServer SC45 484 procs	AlphaServer SC45	Alpha 21264B	484

Preliminary Results Machine List (2 of 2)

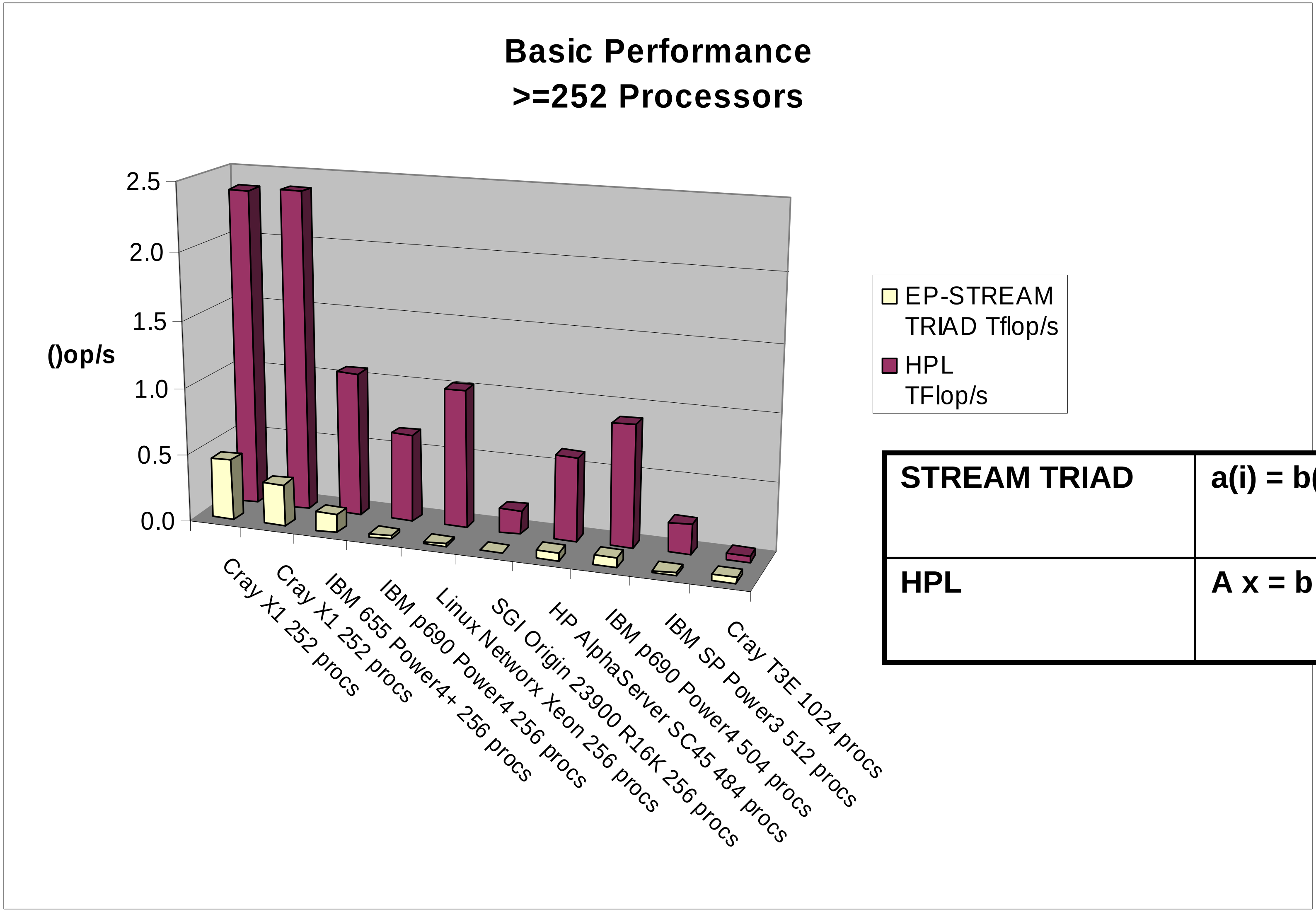
Affiliation	Manufacturer	System	ProcessorType	Procs
IBM	IBM 655 Power4+ 64 procs	eServer pSeries 655	IBM Power 4+	64
IBM	IBM 655 Power4+ 128 procs	eServer pSeries 655	IBM Power 4+	128
IBM	IBM 655 Power4+ 256 procs	eServer pSeries 655	IBM Power 4+	256
NAVO	IBM p690 Power4 504 procs	p690	IBM Power 4	504
ARL	IBM SP Power3 512 procs	RS/6000 SP	IBM Power 3	512
ORNL	IBM p690 Power4 256 procs	p690	IBM Power 4	256
ORNL	IBM p690 Power4 64 procs	p690	IBM Power 4	64
ARL	Linux Networx Xeon 256 procs	Powell	Intel Xeon	256
U Manchester	SGI Altix Itanium 2 32 procs	Altix 3700	Intel Itanium 2	32
ORNL	SGI Altix Itanium 2 128 procs	Altix	Intel Itanium 2	128
U Tenn	SGI Altix Itanium 2 32 procs	Altix	Intel Itanium 2	32
U Tenn	SGI Altix Itanium 2 32 procs	Altix	Intel Itanium 2	32
U Tenn	SGI Altix Itanium 2 32 procs	Altix	Intel Itanium 2	32
U Tenn	SGI Altix Itanium 2 32 procs	Altix	Intel Itanium 2	32
NASA ASC	SGI Origin 23900 R16K 256 procs	Origin 3900	SGI MIPS R16000	256
U Aachen/RWTH	SunFire 15K 128 procs	Sun Fire 15k/6800 SMP-Cluster	Sun UltraSparc III	128
OSC	Voltaire Cluster Xeon 128 procs	Pinnacle 2X200 Cluster	Intel Xeon	128

STREAM TRIAD vs HPL
120-128 Processors



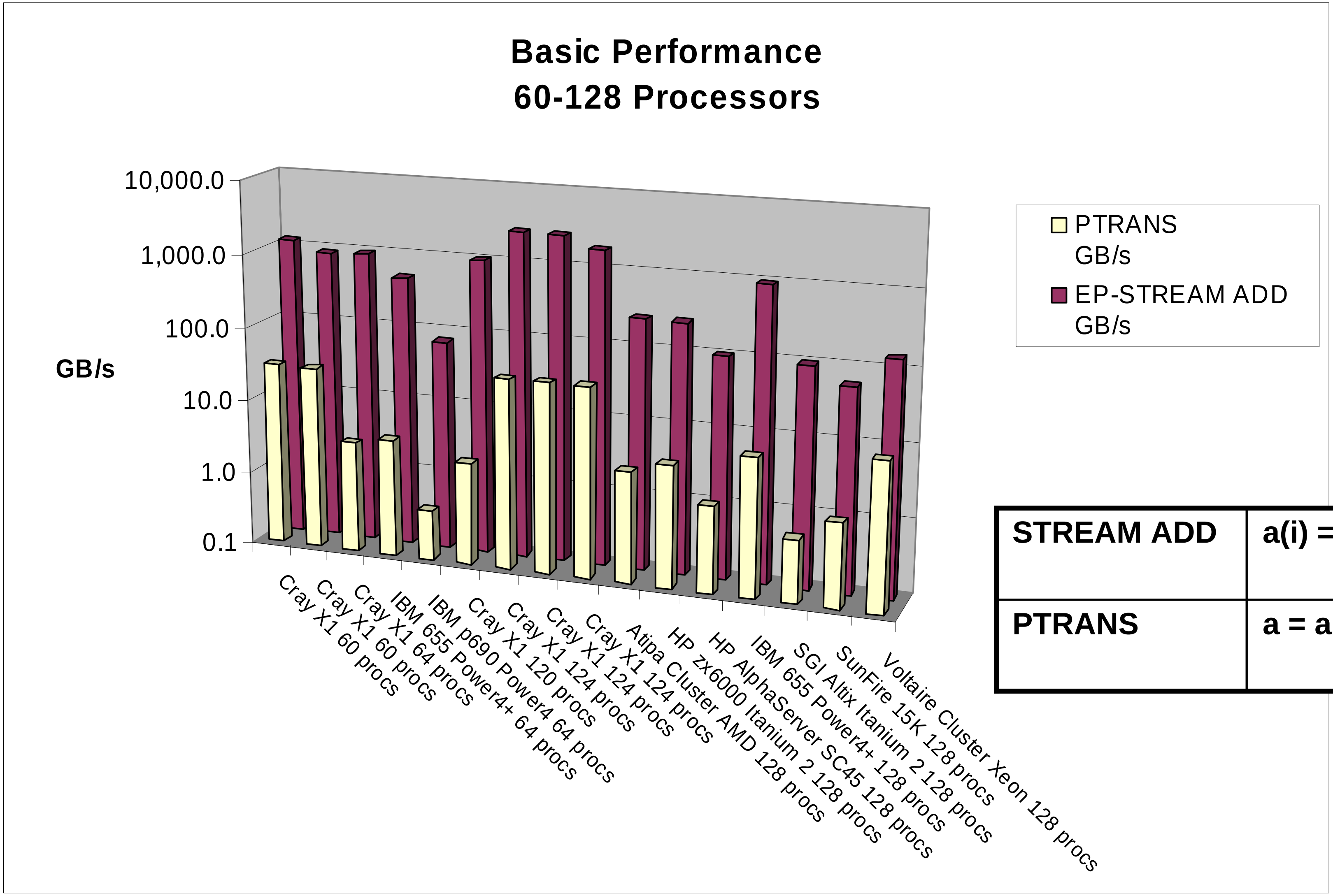
STREAM TRIAD	**a(i) = b(i) + q *c(i)**
HPL	**A x = b**

STREAM TRIAD vs HPL >252 Processors



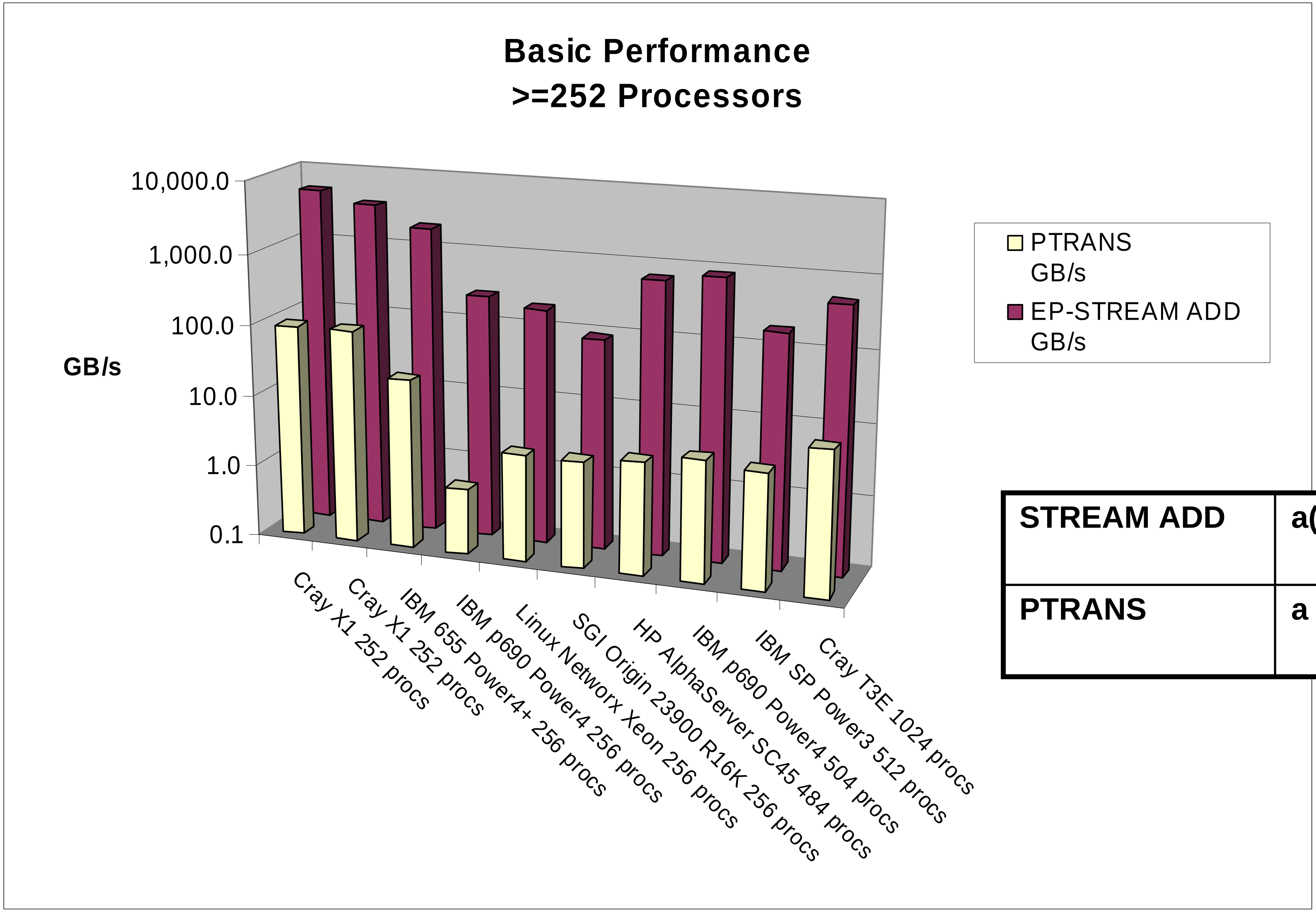
STREAM TRIAD	**a(i) = b(i) + q *c(i)**
HPL	**A x = b**

STREAM ADD vs PTRANS
60-128 Processors



STREAM ADD	**$a(i) = b(i) + c(i)$**
PTRANS	**$a = a + b^T$**

STREAM ADD vs PTRANS
>252 Processors



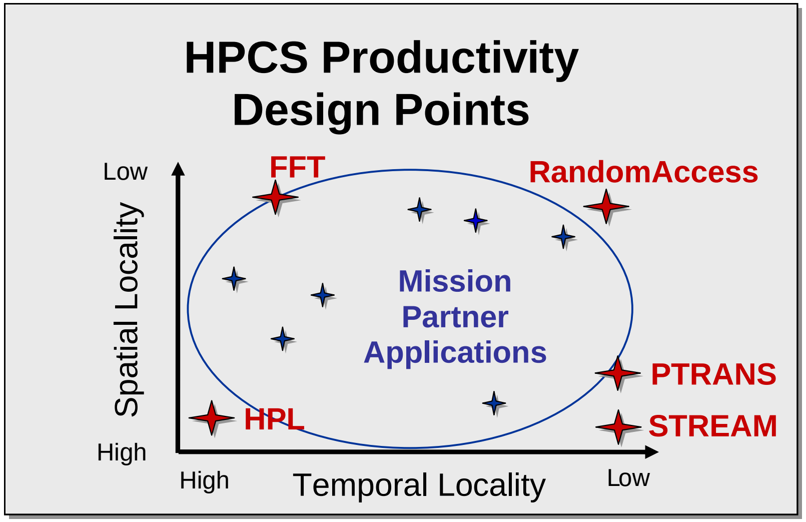
STREAM ADD	**a(i) = b(i) + c(i)**
PTRANS	**a = a + b^T**

Outline

- Brief DARPA HPCS Overview
- Architecture/Application Characterization
- HPCchallenge Benchmarks
- Preliminary Results
- **Summary**

Summary

- DARPA HPCS Subsystem Performance Indicators
  - 2+ PF/s LINPACK
  - 6.5 PB/sec data STREAM bandwidth
  - 3.2 PB/sec bisection bandwidth
  - 64,000 GUPS
- Important to understand architecture/application characterization
  - Where did all the lost “Moore’s Law performance go?”
- HPCchallenge Benchmarks — http://icl.cs.utk.edu/hpcc/
  - Peruse the results!
  - Contribute!


HPCS Productivity Design Points
Low
Spatial Locality
High
FFT
RandomAccess
Mission Partner Applications
PTRANS
STREAM
HPL
High
Temporal Locality
Low